
Subject: CW_PDMENU, changing
Posted by [R. Bauer](#) on Fri, 09 May 1997 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi,

I have created a fine pull down menu.

For example:

File

```
.  
...Open  
.  
...Layout ...* Screen  
.  
... PS file.....idl.ps  
.  
... EPS file....idl.eps  
.  
...Quit
```

The * means that's the default.

I like to set the * to 'PS file' if that was choosen.

How can I do this?

--

R.Bauer

Institut fuer Stratosphaerische Chemie (ICG-1)
Forschungszentrum Juelich
email: R.Bauer@fz-juelich.de

Subject: Re: CW_PDMENU, changing
Posted by [davidf](#) on Fri, 09 May 1997 07:00:00 GMT

Stein Vidar writes in reply to one of my suggestions:

> Ah, a challenge..

How come everyone is taking my suggestions as challenges this week. I am *not* offering any more books!

> Actually, this can be solved more generally without changing
> CW_PDMENU. Below are two procedures, CHECK_MENU and UNCHECK_MENU.

Not only a neat solution, but a recursive procedure to boot!
Nice going.

Cheers!

David

David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 Fax: 970-221-4762
E-mail: davidf@dfanning.com
IDL Programming Tips: <http://www.dfanning.com>

Subject: Re: CW_PDMENU, changing
Posted by [Stein Vidar Hagfors H](#) on Fri, 09 May 1997 07:00:00 GMT

[View Forum Message](#) <> [Reply to Message](#)

David Fanning wrote:

>
> Whoops, in response to R. Bauer's question about changing
> buttons in a pull-down menu I wrote:
>
>> Widget_Control, screenButtonID, Set_Value="Screen"
>> Widget_Control, psFileButtonID, Set_Value="* PS file"
>
> After I posted I read the Subject line and realized he is
> using CW_PDMENU for his pull-down menus. I've sort of
> given up on CW_PDMENU since I have never been successful
> explaining how it works to anybody. :(I always just
> create my own menus using WIDGET_BUTTON and the MENU
> keyword.
>

I do agree that the CW_PDMENU is only making things more

complicated than using the straightforward `widget_button` approach.

- > So my advice above will only work if `CW_PDMENU` is
- > modified to return the ID's of the buttons it creates.
- > It would be a very good idea if these IDs were returned
- > in a `BUTTON_IDS` keyword, for example.

Ah, a challenge..

Actually, this can be solved more generally without changing `CW_PDMENU`. Below are two procedures, `CHECK_MENU` and `UNCHECK_MENU`.

`CHECK_MENU,MENU_ID,CHECK_UVALUE`

will check off all items in the menu rooted at `MENU_ID` with an asterisk + space in front of the button text. Optionally, a different string can be used to check off the buttons.

`UNCHECK_MENU,MENU_ID` will uncheck all items in the menu rooted at `MENU_ID` by chopping off any leading `CHECKSTRING` from the button text (default `"* "`).

If called with a second argument, `CHECK_UVALUE`, only buttons with the matching uvalue will be unchecked - leaving the possibility of controlling an options on/off menu. I haven't tested that mode of operations, though.

I also wrote a simple test program (`CHECKMENU`) to demonstrate it's use. It's quite ok to construct the menu using `CW_PDMENU` if you like, of course.

Cheers,

Stein Vidar H. Haugan

```
PRO check_menu,menu,check_uvalue,checkstring=checkstring
```

```
  IF n_elements(checkstring) EQ 0 THEN checkstring = "* "
```

```
  widget_control,menu,get_uvalue=this_uvalue
```

```
  szc = size(check_uvalue)
```

```
  szu = size(this_uvalue)
```

```
  IF szu(szu(0)+1) EQ szc(szc(0)+1) THEN BEGIN
```

```
    IF this_uvalue EQ check_uvalue THEN BEGIN
```

```
      widget_control,menu,get_value=val
```

```

    widget_control,menu,set_value=checkstring + val
  END
END
child = widget_info(menu,/child)
IF child NE 0L THEN
check_menu,child,check_uvalue,checkstring=checkstring

  sibling = widget_info(menu,/sibling)
  IF sibling NE 0L THEN
check_menu,sibling,check_uvalue,checkstring=checkstring
END

PRO uncheck_menu,menu,check_uvalue,checkstring=checkstring

  IF n_elements(checkstring) EQ 0 THEN checkstring = "* "

  IF n_params() EQ 2 THEN BEGIN
    szc = size(check_uvalue)
    szu = size(this_uvalue)
    IF szu(szu(0)+1) EQ szc(szc(0)+1) THEN BEGIN
      IF this_uvalue EQ check_uvalue THEN BEGIN
        widget_control,menu,get_value=val

widget_control,menu,set_value=strmid(val,strlen(checkstring) ,1000)
      END
      END
    END ELSE BEGIN
      widget_control,menu,get_value=val
      IF strpos(val,checkstring) EQ 0 THEN BEGIN

widget_control,menu,set_value=strmid(val,strlen(checkstring) ,1000)
      END
      END

child = widget_info(menu,/child)
sibling = widget_info(menu,/sibling)

  IF n_params() EQ 2 THEN BEGIN
    IF child NE 0L THEN uncheck_menu,child,check_uvalue,$
      checkstring=checkstring
    IF sibling NE 0L THEN uncheck_menu,sibling,check_uvalue,$
      checkstring=checkstring
  END ELSE BEGIN
    IF child NE 0L THEN uncheck_menu,child,checkstring=checkstring
    IF sibling NE 0L THEN uncheck_menu,sibling,checkstring=checkstring
  END
END

```

```
PRO checkmenu_event,ev

widget_control,ev.top,get_uvalue=status
widget_control,ev.id,get_uvalue=uvalue

uncheck_menu,status.menu

check_menu,status.menu,uvalue

IF uvalue EQ "QUIT" THEN widget_control,ev.top,/destroy

END
```

```
PRO checkmenu

base = widget_base(/column)

menu = widget_button(base,value='Menu',menu=2)

dummy = widget_button(menu,value='Item a', uvalue='A')
dummy = widget_button(menu,value='Item b',uvalue='B')
sub = widget_button(menu,value='Other..',menu=1)
dummy = widget_button(sub,value='Item c',uvalue='C')
dummy = widget_button(sub,value='Item d',uvalue='D')
dummy = widget_button(menu,value='Item z',uvalue='Z')

dummy = widget_button(menu,value='Quit',uvalue='QUIT')

status = {menu:menu}

widget_control,base,set_uvalue = status

widget_control,base,/realize

xmanager,'checkmenu',base

xmanager
END
```

Subject: Re: CW_PDMENU, changing
Posted by [agrap](#)s on Fri, 09 May 1997 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

"R. Bauer" <r.bauer@fz-juelich.de> writes:

- > I have created a fine pull down menu.
- > The * means that's the default.
- > I like to set the * to 'PS file' if that was choosen.
- > How can I do this?

It would be neat if there were a "checked" option of the menu items.
 I manually check the UVALUE in my event loop and check and uncheck the menu items based on that. There's probably a fancier way.

But, here is an example of what I do manually with checking and unchecking menu items.

```
;Create Diagnostics Menu
switch_menu = WIDGET_BUTTON(bar_base, Value='Diagnostics', /Menu)
switch_btn3 = WIDGET_BUTTON(switch_menu, Value='*Time Simulation', $
Uvalue='*Time Simulation')
switch_btn4 = WIDGET_BUTTON(switch_menu, Value='*Beep When Done', $
Uvalue='*Beep When Done')
```

Then part of my event loop looks like:

```
CASE eventval OF
; These menu items act like a toggle switch using '*'

'*Time Simulation': BEGIN
WIDGET_CONTROL, event.id, set_value = 'Time Simulation', $
set_uvalue = 'Time Simulation'
END ; '*Time Simulation'
'*Beep When Done': BEGIN
WIDGET_CONTROL, event.id, set_value = 'Beep When Done', $
set_uvalue = 'Beep When Done'
END ; '*Beep When Done'
'Time Simulation': BEGIN
WIDGET_CONTROL, event.id, set_value = '*Time Simulation', $
set_uvalue = '*Time Simulation'
END ; 'Time Simulation'
'Beep When Done': BEGIN
WIDGET_CONTROL, event.id, set_value = '*Beep When Done', $
set_uvalue = '*Beep When Done'
END ; 'Beep When Done'
ENDCASE
```

viel Glueck!

Amara

--

Amara Graps email: agraps@netcom.com
Computational Physics vita: finger agraps@best.com
Multiplex Answers URL: http://www.amara.com/

Subject: Re: CW_PDMENU, changing
Posted by [davidf](#) on Fri, 09 May 1997 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Whoops, in response to R. Bauer's question about changing buttons in a pull-down menu I wrote:

```
> Widget_Control, screenButtonID, Set_Value="Screen"  
> Widget_Control, psFileButtonID, Set_Value="* PS file"
```

After I posted I read the Subject line and realized he is using CW_PDMENU for his pull-down menus. I've sort of given up on CW_PDMENU since I have never been successful explaining how it works to anybody. :(I always just create my own menus using WIDGET_BUTTON and the MENU keyword.

So my advice above will only work if CW_PDMENU is modified to return the ID's of the buttons it creates. It would be a very good idea if these IDs were returned in a BUTTON_IDS keyword, for example.

Cheers!

David

David Fanning, Ph.D.
Fanning Software Consulting
Customizable IDL Programming Courses
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com

Subject: Re: CW_PDMENU, changing
Posted by [davidf](#) on Fri, 09 May 1997 07:00:00 GMT

R. Bauer <r.bauer@fz-juelich.de> writes:

- > I have created a fine pull down menu.
- > For example:
- >
- > File
- > .
- > ...Open
- > .
- > ...Layout ...* Screen
- > .
- > PS file.....idl.ps
- > .
- > EPS file....idl.eps
- > .
- > ...Quit
- >
- > The * means that's the default.
- >
- > I like to set the * to 'PS file' if that was chosen.
- > How can I do this?

Widget_Control, screenButtonID, Set_Value="Screen"
Widget_Control, psFileButtonID, Set_Value="* PS file"

Cheers!

David

David Fanning, Ph.D.
Fanning Software Consulting
Customizable IDL Programming Courses
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: <http://www.dfanning.com>

Subject: Re: CW_PDMENU
Posted by [davidf](#) on Tue, 13 Apr 1999 07:00:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

Pavel Romashkin (pavel@terra.colorado.edu) writes:

- > Have you noticed that, when creating widget applications with top-level
- > pulldown menus, if you use CW_PDMENU for creating menus, then there's no
- > way to use the 1st child of the top base (which is the menu bar base) to

> carry the widget STATE information, because UVAL of all CW_PDMENUs and
> the menu bar are reset to the last CW_PDMENU's UVALUE? I have asked
> RSInc about it and they said this is because CW_PDMENU uses the menu bar
> as a base (since menu bar can only have buttons as children). They said
> they are not likely to alter this behavior for the sake of backward
> compatibility. I worked around this by using WIDGET_BUTTONS on the menu
> bar, and populating them with CW_PDMENUs. How did you get around this?

I get around it by not using CW_PDMENU at all. I just create buttons. This gives me great flexibility in assigning event handlers, moving buttons around in my interface, adding new buttons, etc. If I want new functionality in my program I just create a new button, assign it an event handler, write the event handler and I am done with it. I don't have to worry about breaking code I've already written, etc.

If I want one event handler to handle all the button events in a pull-down menu, I just assign the event handler to the button at the top of the button hierarchy. The sub-button events "bubble up" to the correct event handler.

Cheers,

David

--

David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
Toll-Free IDL Book Orders: 1-888-461-0155
